

```

import React, { useState } from "react";

const ToDoList = () => {
  const [addInput, setAddInput] = useState("");
  const [items, setItems] = useState([]);
  const [isAddBtn, setIsAddBtn] = useState(true);
  const [searchInput, setSearchInput] = useState("");
  const [isEditItem, setIsEdititem] = useState(null);

  const addListItem = () => {
    if (!addInput) {
      alert("please Fill input");
    } else if (addInput && !isAddBtn) {
      let updatedItem = [...items];
      const itemIndexToEdit = updatedItem.findIndex(
        (item) => item.id === isEditItem
      );
      if (itemIndexToEdit !== -1) {
        updatedItem[itemIndexToEdit].name = addInput;
        setItems(updatedItem);
        setIsAddBtn(true);
        setAddInput("");
        setIsEdititem(null);
      }

      // setItems(
      //   items.map((elem) => {
      //     if (elem.id === isEditItem) {
      //       return { ...elem, name: searchInput };
      //     }
      //     return elem;
      //   })
      // );
    } else {
      const allInputele = {
        id: new Date().getTime().toString(),
        name: addInput,
      };
      setItems([...items, allInputele]);
      setAddInput("");
    }
  };

  const deleteItems = (index) => {
    const updateItems = items.filter((currEle) => {

```

```

    return index !== currEle.id
  });
  setItems(updateItems);
};

const editItems = (name, index) => {
  setIsAddBtn(false);
  setAddInput(name);
  setIsEdititem(index);
};

return (
  <>
    <div className="container">
      <div className="todoList">
        <h1>ToDo List</h1>
        <input
          type="text"
          id="addInput"
          placeholder="Add to List"
          value={addInput}
          onChange={(event) => setAddInput(event.target.value)}
        />

        <button id="btn" onClick={addListItem}>
          {isAddBtn ? "Add" : "Save"}
        </button>

        <ol id="todoList">
          {items.map((itemval, index) => {
            return (
              <li key={index}>
                <span>{itemval.name}</span>
                <span className="update">
                  <i
                    className="fa fa-edit"
                    onClick={() => {
                      editItems(itemval.name, itemval.id);
                    }}
                    aria-hidden="true"
                  ></i>
                </span>
                <span className="delete">
                  <i
                    className="fa fa-trash"

```

```

        onClick={() => {
            deleteItems(itemval.id);
        }}
        aria-hidden="true"
    ></i>
</span>
</li>
    );
    })}
</ol>
</div>

{/* <div className="searchList">
    <h1>Search from List</h1>
    <input
        type="text"
        id="searchInput"
        placeholder="Search"
        onkeyup="searchString()"
    />

    <ol id="searchList"></ol>
</div> */}
</div>
</>
);
};
export default ToDoList;

*{
    padding: 0;
    margin: 0;
    box-sizing: border-box;
}
body{
    background-color: rgb(115, 166, 201);
}
.container{
    display: flex;
    align-items: center;
    justify-content: center;
    gap: 20px;
    margin-top: 100px;
    color: white;

```

```
}
.todoList{
  padding-top: 20px;
  width: 23rem;
  min-height: 30rem;
  background-color: rgb(123, 110, 209);
  text-align: center;
  border-radius: 20px;
}
.todoList input{
  margin-top: 10px;
  padding: 5px 10px;
  height: 40px;
  width: 250px;
  border-radius: 20px;
  border: none;
}
.todoList button{
  padding: 12px;
  border-radius: 10px;
  border: none;
  background-color: rgb(40, 26, 234);
  color: white;
  cursor: pointer;
  width: 60px;
  font-weight: 600;
  margin-left: 10px;
}
.todoList ol{
  margin-top: 20px;
}
.todoList ol li{
  position: relative;
  text-align: left;
  margin: 0 15px 0 40px;
  overflow: hidden;
  line-height: 2;
}
.mylist li{
  margin: 0 40px;
}
ol li .delete{
```

```
position: absolute;
right: 20px;
width: 30px;
height: 30px;
border-radius: 50%;
cursor: pointer;
text-align: center;
}

ol li .update{
position: absolute;
right: 60px;
cursor: pointer;
width: 30px;
height: 30px;
border-radius: 50%;
text-align: center;
}

ol li .Lvalue{
width: 50%;
background-color: aqua;
}

ol li .update:hover{
background-color: green;
color: white;
}

ol li .delete:hover{
background-color: red;
color: white;
}

#searchList {
margin: 20px 100px;
}

.searchList{
padding-top: 20px;
width: 23rem;
min-height: 30rem;
background-color: white;
text-align: center;
border-radius: 20px;
}

.searchList input{
margin-top: 10px;
```

```
padding: 10px;
width: 250px;
border-radius: 13px;
}
.searchList button{
padding: 12px;
border-radius: 10px;
border: none;
background-color: blue;
color: white;
cursor: pointer;
width: 60px;
font-weight: 600;
}
```